KARL HEINZ BORGWARDT

AVERAGE COMPLEXITY OF A GIFT-WRAPPING ALGORITHM FOR DETERMINING THE CONVEX HULL OF RANDOMLY GIVEN POINTS

Abstract

This paper presents an algorithm and its probabilistic analysis for constructing the convex hull of m given points in R^n, the n-dimensional Euclidean space. The algorithm under consideration combines the Gift-Wrapping-concept with the so-called Throw-Away-Principle for nonextremal points. The latter principle had been used for a convex-hull-construction algorithm in R^2 and for its probabilistic analysis in a recent paper by Borgwardt, Gaffke, Jünger and Reinelt [Borgwardt, et.al(1991)]. There, the considerations remained much simpler, because in R^2 the construction of the convex hull essentially requires recognition of the extremal points and of their order only.

In this paper the Simplex-Method is used to organize a walk over the surface of the convex hull. During this walk all facets are discovered. Under the condition of general position this information is sufficient, because the whole face-lattice can simply be deduced when the set of facets is available.

Exploiting the advantages of the revised Simplex-Method reduces the update-effort to an nxn–matrix and the number of calculated quotients for the pivot-search to the points which are not thrown away.

For this algorithm a probabilistic analysis can be carried out. We assume that our m random points are distributed identically, independently and symmetrically under rotations in R^n. Then the calculation of the expected effort becomes possible for a whole parametrical class of distributions over the unit ball. The results mean a progress in three directions

- a parametrization of the expected effort can be given
- the dependency on n – the dimension of the space – can be evaluated
- the additional work of preprocessing for detecting the vertices can be avoided without losing its advantages.

1. Introduction

The Problem and its Geometry

Let m points $a_1,...,a_m \in \mathbb{R}^n$ be given. We want to construct or determine the convex hull of these points, which is denoted by $Y := CH(a_1,...,a_m)$.
In general, we base our considerations on the following

Condition of Nondegeneracy

(1.1) Each subset of n vectors out of $\{a_1,...,a_m\}$ is linearly independent and each subset of $n+1$ such points is in general position.

It will be our aim to study the expected number of arithmetical operations for determining the convex hull under several specified stochastic models, i.e. assumptions on the distribution of the data $a_1,...,a_m$. Here we concentrate on distributions where our condition of nondegeneracy is satisfied almost surely, i.e. with probability 1. Since the computational effort is bounded for the treatment of m points, it is permitted to ignore the degeneracy-cases completely for the purpose of calculating the expected values.

In addition, our condition of nondegeneracy makes sure that the polyhedron Y becomes simplicial. Hence all facets of Y are n–1-dimensional boundary simplices. All the other (lower-dimensional) faces of Y can simply be obtained by dropping some of the points generating a certain facet.
For instance, let $a_1,...,a_n$ generate a facet, i.e. $CH(a_1,...,a_n)$ is a facet. Then $CH(a_1,...,a_k)$ (with $1 < k < n$) is a k–1-dimensional face of Y and every face can be obtained in that way. Now it is clear that knowing the facet-set is sufficient.

Hence we are allowed to formulate a "reduced task":

(1.2) Determine all facets of Y.

We shall denote the set of Y-facets by F.

The author has studied such polytopes $CH(a_1,...,a_m)$ thoroughly in [Borgwardt (1987)] for the purpose of calculating the number of iteration steps required by the Simplex-Algorithm as well from a geometrical and from a probabilistic point of view. There the Simplex-Algorithm moved on a path over successively adjacent facets from a start facet to a target or final facet. This time, we move in the same way, but with the aim to visit every facet of Y.

Before discussing some popular concepts for constructing convex hulls (including our own), let us summarize some important geometrical properties of the surface of Y.

Definition 1.1

Let Δ be an n-tupel $(\Delta^1,...,\Delta^n)$ with components $\Delta^i \in \{1,...,m\}$ for i=1,...,n and $\Delta^i < \Delta^{i+1}$ for i=1,...,n–1.
Further let F be the set of facets of Y, which is the set of all $(n-1)$-dimensional faces. Accordingly, V shall denote the set of all vertices or extremal points of Y. It is intuitively clear that $\mathrm{V} \subset \{a_1,...,a_m\}$.
By #F and #V we shall denote the cardinalities (number of elements) of the corresponding sets.In [Borgwardt (1987)] the following facts have been proven.

Lemma 1.1

a) Every facet is a boundary simplex (BS) of the type $CH(a_{\Delta^1},...,a_{\Delta^n})$, which is generated by n points $a_{\Delta^1},...,a_{\Delta^n}$.

b) Every boundary simplex itself is bounded by its n different n–2-dimensional faces, which we call side of a BS or border of a boundary simplex. That means that a BS $CH(a_{\Delta^1},...,a_{\Delta^n})$ possesses the n borders (BSB) $CH(a_{\Delta^1},...,a_{\Delta^{n-1}}),...,CH(a_{\Delta^1},...,a_{\Delta^{i-1}},a_{\Delta^{i+1}},...,a_{\Delta^n}),...,CH(a_{\Delta^2},...,a_{\Delta^n})$.

c) To each border of a boundary simplex (BSB) exactly two boundary simplices or facets are incident. That means:
If $CH(a_{\Delta^1},...,a_{\Delta^{i-1}},a_{\Delta^{i+1}},...,a_{\Delta^n})$ is a BSB, then it can be augmented with two different points a_{i_1} and a_{i_2} such that $CH(a_{\Delta^1},...,a_{\Delta^{i-1}},a_{i_1},a_{\Delta^{i+1}},...,a_{\Delta^n})$ and

$CH\left(a_{\Delta^1},...,a_{\Delta^{i-1}},a_{i_2},a_{\Delta^{i+1}},...,a_{\Delta^n}\right)$ are facets of Y. We call a_{i_1} and a_{i_2} augmenting points for the BSB.

d) If boundary-simplex-borders of two different facets are identical, e.g. $CH\left(a_{\Delta^1},...,a_{\Delta^{i-1}},a_{\Delta^{i+1}},...,a_{\Delta^n}\right)$, then this is a BSB separating the two facets, and the corresponding augmenting points determine a unique facet $CH\left(a_{\Delta^1},...,a_{\Delta^i},...,a_{\Delta^n}\right)$ each.

Some famous algorithmic concepts

Relying on the thesis of Dwyer [Dwyer (1988a)] we briefly describe the most important algorithmical approaches.
At a first glance it might be desirable to know the set of vertices in advance, which can be achieved by so-called preprocessing, and only afterwards one could apply the convex-hull-construction on V instead of $\{a_1,...,a_m\}$. Whether such a preprocessing is worth while can be judged only for the single case and the special algorithm.

For each point a_i we can decide whether it is a vertex of Y by solving the following linear optimization problem:

(1.3) minimize $\bar{a}^T x$

$$\begin{aligned} \text{subject to} \quad & (a_j - \bar{a})^T x \le 1 \quad \text{for } j=1,...,m, \quad j \ne i \\ & (a_i - \bar{a})^T x = 1 \end{aligned}$$

where $\bar{a}$ stands for $\frac{1}{m}(a_1+...+a_m)$, i.e. $\bar{a}$ is the barycenter of Y.

If there are feasible points x, then a_i must be a vertex ($\bar{a}$ cannot be a vertex as long as the problem is nondegenerate).

So we have to solve an LP with m restrictions and n variables. From the work of Megiddo [Megiddo(1984)],Clarkson [Clarkson (1986)] and Dyer and Frieze [Dyer and Frieze (1989)] it is known that the worst-case complexity for such a problem is bounded by $O(m) \cdot C(n)$.
Unfortunately $C(n)$ will not be polynomial, on the other side the dependency on m is linear. But this should not induce the impression of very quick solvability, notice the bad behaviour of $C(n)$.

On the other hand the exclusion of the irrelevant a_i's may dramatically reduce the computation time for the pure convex-hull-construction. Whether this is the case or not, is part of our study.

Now some examples for hull-construction-methods follow.

a) Enumeration

Combinatorially every possible n-tupel $(a_{\Delta^1},...,a_{\Delta^n})$ is checked for the property of generating a facet. For this purpose one calculates the normal vector on the hyperplane induced by the n selected points. This hyperplane (affine hull of $a_{\Delta^1},...,a_{\Delta^n}$) divides R^n into two halfspaces. Now it is easy to check whether one of these two halfspaces contains all points $a_1,...,a_m$. This would be equivalent to $CH(a_{\Delta^1},...,a_{\Delta^n})$ being a facet. The algorithm is extremely lengthy and time-wasting. Its worst-case-complexity is $O(m^n)$.

b) Beneath-Beyond-Algorithm of Kallay and Seidel

Here it is necessary to keep an updated file of all facets and boundary-simplex-borders (BSB) for the auxiliary polytope $CH(a_1,...,a_l)$ $(l \in \{n,...,m\})$. In a main iteration step the point a_{l+1} is accepted and integrated. Starting from $CH(a_1,...,a_l)$ one determines $CH(a_1,...,a_{l+1})$. Both lists are updated.
In the intermediate steps of the iteration we must decide which of the old facets remain, which disappear and which enter now. Therefore information on the relative position of the new point a_{l+1} to the affine hull of an old facet is required. Since we look at a facet of $CH(a_1,...,a_l)$, all of the remaining points had assembled in one of the two halfspaces. Now it is up to a_{l+1} to decide whether it belongs to the same halfspace (hereby confirming that the old facet still holds) or to the opposite halfspace, in which case the old facet is dropped.
To the old facets still in the game we add new facets in the following way. Determine to each destroyed old facet those facet-borders (BSB) which did simultaneously belong to a still existing facet and a dropped facet. After that augment the BSB (generated by $n-1$ points) with a_{l+1}. Hereby one creates exactly the new facets. In our files the changes must be recorded accordingly.
This algorithm does not admit a simple analysis, because the facet-sets in intermediate steps may vary dramatically. In addition, it depends strongly on the chosen order of

$a_1,...,a_m$, and the auxiliary polytope hardly has a strong similarity with the final and desired facet-set of $CH(a_1,...,a_m)$.

c) Shelling-Algorithm of Seidel

This concept is best describable by giving an example of physics. One tries to simulate the lightening effect of a lamp, which shines on a convex body (polytope), and studies the change of visibility when the lamp disappears from the body or when it approachest he body. Starting from the barycenter $\bar{a}$, one moves the lamp on the axis $\bar{a}+\lambda e_n$ ($\lambda \in$ R, $e_n = (0,...,0,1) \in$ R^n), with λ starting from 0 and increasing to infinity. Now imagine $CH(a_1,...,a_m)$ as our convex body. First we observe the order in which $a_1,...,a_m$ become visible or light. Mathematically this means that the interval $[a_i, \bar{a}+\lambda e_n]$ does not intersect the interior of Y. The intersection-question can be answered by solving the following linear optimization problem.

(1.4) maximize λ

$$\text{subject to} \quad \sum_{\substack{j=1 \\ j\infty i}}^{m} \rho_j\,(a_j - a_i) = \bar{a} + \lambda e_n$$

$$\rho_1,...,\rho_m \geq 0, \lambda \geq 0$$

If there are feasible λ's, then their maximum delivers that λ, where our connection interval leaves the recession cone of Y at a_i.
From the solution we obtain $n-1$ points a_j with $\rho_j > 0$, while all other ρ's are 0. The selected corresponding points (together with a_i) generate a facet, which becomes visible simultaneously with a_i. By the way we associate with each a_i a value λ_i (visibility-time) and thus another ordering of some points. But so far we have moved only on one side of the polytope (let's say the north side). Only the upper half is lightened so far. Now we consider the analoguous movement on the opposite side (south side), but we count vice versa, (i.e. from –` to 0). So we complete our point-ordering. The physical effect now describes a continuous north-south movement of a horizon over the polytope, starting in the north, moving to the equator and beyond until we reach the south pole.

We must be aware that so far only some of the facets have been listed together with their visibility-point. We have to provide a list of all observed facets, BSB's and of their respective sides or borders (n–3-dimensional simplices).

Of interest are such BSB's which are incident to both a visible and an unvisible facet. The set of all such BSB's forms the so-called visibility-horizon on Y.
Assume that all visible facets have been listed, then for the next visible facet there are two cases. Either it is generated completely by points which are already visible. Then we obtain it by joining such an n–3-simplex with two incident n–2-simplices (BSB's). Or we need <u>one</u> absolutely new point to augment an available BSB.
For the first case one needs a file of all critical n–3-simplices (those adjacent to a non-visible facet).
In the second case we have exactly that facet, which had before accompanied a certain point a_i in becoming visible.

For all points and all facets, BSB's and n–3-faces one needs a list of visibility times.
The visibility-time of point a_i was calculated in (1.4), the corresponding time for a facet is simply calculated by intersecting its affine hull with the axis $\bar{a}+\lambda e_n$. Updating of the mentioned files is no severe problem.
One needs #F main iterations for discovering new facets. The updating effort can be bounded from above by $\#\mathrm{F} \cdot C_1(n) \cdot \log m$, whereas solution of the m necessary LP's makes an effort of $m \cdot m \cdot C_2(n)$.
A possibility for saving would come from knowledge and use of V instead of $\{a_1,...,a_m\}$. Then m^2 could be replaced by $(\#\mathrm{V})^2$.
The key to such an improvement lies in Preprocessing combined with a Divide-and-Conquer-method due to Bentley and Shamos [Bentley and Shamos (1978)] for the vertex-enumeration. For that purpose we divide the point-set in two halves, determine the vertices of the convex hull for each half, and decide via linear optimization which of those vertices are vertices of the convex hull of the joint set.
Applying this concept recursively becomes extremely profitable, if we obtain a considerable share of redundant points in the single steps.

We can observe the saving effect by looking at the order of the expected effort under certain stochastic models. Here Shelling without Preprocessing requires (as a function of m) an effort of

(1.5) $O(m^2 + E(\#\,\mathrm{F}) \cdot \log m)$

and with the help of Preprocessing

(1.6) $O(m + E(\#9)^2 + E(\#\text{F}) \log \text{m})$

For the distributions discussed in this paper later, we obtain in the case of parameter k for the Shelling-Algorithm an expected m-dependency of

(1.7) $O(m)$ for $k > \frac{n-1}{2}$

(1.8) $O(m \log m)$ for $k = \frac{n-1}{2}$

(1.9) $O\left(m^{\frac{2(n-1)}{2k+n-1}}\right)$ for $k < \frac{n-1}{2}$

when we use Preprocessing.

Without Preprocessing even the expected value stays at $O(m^2)$.

d) Gift-Wrapping-Algorithm of Chand and Kapur

Analoguous statements can be given for the Gift-Wrapping-Algorithm, which will be the subject of our study.
This algorithm realizes a walk over the surface of Y from facet to (adjacent) facet. Each facet-change can be interpreted as crossing a BSB. Our goal is to discover all facets. But sometimes our walk may lead into a dead end, i.e. a facet whose facet-neighbours all are already known. In this case one has to walk back until a facet with unknown neighbours is reached.

Again, the number of main iterations is $O(\#\text{F})$. Since a facet-change crosses a BSB which belongs to both facets, we essentially replace one generator-point by another generator. But this substitute is unique. So for each of the $m–n$ remaining points it has to be checked whether it is the searched substitute. The updating effort for the necessary data-files amounts to $O(\#\text{F} \cdot \log(\#\text{F}))$ as a function of m only. Dwyer [Dwyer (1988a)] obtains for the expected effort statements like

(1.10) $O(m \cdot E(\#\mathrm{F}))$ for Gift-Wrapping without Preprocessing

and

(1.11) $O(m + E(\# 9)^2 + \mathrm{E}(\#9 \cdot \#\mathrm{F}))$ with Preprocessing.

Our aim is to calculate the expected complexity

– for a whole family of certain distributions over the unit ball, which is parametrized, and the result should show the dependency on that parameter,
– exactly also in its dependency upon n, the dimension of the space,
– for an algorithm avoiding Preprocessing in order to get along without the very bad influence of solving many LP's and without wasting time for a possibly useless job,
– showing that the success of Preprocessing with respect to the m-dependency can also be achieved by our algorithm, which saves time implicitly.

Therefore we develop a rule how to exclude a lot of points from the substitute-check for each individual facet. This is a kind of a Throw-Away-Principle.
Finally we achieve the following main result.

<u>Theorem 1</u>

For rotation-symmetric, independent and identical distribution of m random points on an n-dimensional unit ball with arbitrary parameter k [(–1, ∞) and radial density function

(1.12)
$$f_k(r) = \frac{(1-r^2)^k r^{n-1}}{\int_0^1 (1-\tau^2)^k \tau^{n-1} d\tau}$$

our combination of Gift-Wrapping-Algorithm with Throw-Away-Principle requires on the average not more than

(1.13)
$$C \cdot \{ \; m\, C_1(n,k) + m^{\frac{n-1}{n+1+2k}} (\ln m) \; C_2(n,k) + m^{1+\frac{n-3-2k}{n+1+2k}} \; C_3(n,k) +$$

$$+ \; m^{1-\frac{2k+2}{n+1+2k}} \; (\ln m)^{\frac{2k+2}{n+1+2k}+1} \; C_4(n,k) \}$$

arithmetic operations for calculating all facets of $CH(a_1,...,a_m)$.
The functions $C_1(n,k),...,C_4(n,k)$ can be given explicitly.

e) Gift-Wrapping without storage by Avis and Fukuda

Avis and Fukuda [Avis and Fukuda (1992)] developed a version of the Gift-Wrapping-Algorithm, which avoids most of the storage requirements. This can be done by exploiting the fact, that in the nondegeneracy case the facet-enumeration problem can be dualized to a vertex-enumeration problem on the feasible polyhedron of a linear optimization problem. Using a fixed variant of the Simplex-Method (e.g. Bland's Rule), one can start at each vertex, running along a Simplex Path improving the objective, until the optimal vertex is reached. The progress directions depend only on the current vertex, not on the place where we started. So the set of all Simplex Paths generated by that variant creates a tree with root at the optimal vertex. For doing Gift-Wrapping this can be used for showing the way for enumeration. One simply runs along that tree with depth-first-search. At each vertex we have to figure out which of the possible pivot steps are so-called "reverse optimization steps", i.e. which pivot steps are done in reverse direction on such an optimization path. This enables us to run down to the leaves of the tree. For getting upward again, one applies the variant, which leads the same way up until we meet a parent-vertex allowing additional reverse variant-steps. Those can easily be found and identified as non-used steps.
The advantage lies in the fact that we need not search for the possible non-used steps in a file of previous vertices or edges, hence we save the storage effort. The worst-case-effort for that procedure turns out to be

(1.14) $\qquad O(m \cdot n \cdot \#\mathrm{F})$.

2. The role of the Simplex-Algorithm in a walk over the surface of Y.

Similarly to the procedure for solving linear optimization-problems, the Simplex-Algorithm can be used to organize a walk over the surface of Y, which eventually visits all facets of Y.
According to Lemma 1.1 every BS has exactly n exits, each BSB stands for such an exit resp. for the border between two facets. In [Borgwardt (1987)] we have described how to walk over successively adjacent boundary simplices, where each simplex-change corresponds to a pivot step.
If we have the intention to leave the facet $CH(a_{\Delta^1},\ldots,a_{\Delta^n})$ via the exit BSB $CH(a_{\Delta^1},\ldots,a_{\Delta^{n-1}})$, then there are $m-n$ candidates available for replacing $a_{\Delta^n} =: a_{i_1}$. Exactly one of them (w.l.o.g. a_{i_2}) is actually able to augment $(a_{\Delta^1},\ldots,a_{\Delta^{n-1}})$ in such a way that $CH(a_{\Delta^1},\ldots,a_{\Delta^{n-1}},a_{i_2})$ becomes a facet or boundary simplex.
This means that $a_{\Delta^1},\ldots,a_{\Delta^{n-1}},a_{i_1}$ span a hyperplane $H(a_{\Delta^1},\ldots,a_{\Delta^{n-1}},a_{i_1})$, which divides R^n in such a way that all points $a_1,\ldots,a_m$ lie in one of the two closed halfspaces induced by H.
If we interpret the linear independent vectors $a_{\Delta^1},\ldots,a_{\Delta^n}$ as a basis of R^n, then $a_1,\ldots,a_m$ as well as $e_1,\ldots,e_n$ are generated as linear combinations of the basis-vectors.
Let us denote by A_Δ the submatrix of $A = (a_1,\ldots,a_m)$ consisting of the columns $a_{\Delta^1},\ldots,a_{\Delta^n}$.
Then the formulas

$$\begin{aligned} &\alpha_1 = (A_\Delta)^{-1}a_1, \ldots ,\alpha_m = (A_\Delta)^{-1}a_m, \\ &\gamma_1 = (A_\Delta)^{-1}e_1, \ldots ,\gamma_n = (A_\Delta)^{-1}e_n \end{aligned} \tag{2.1}$$

deliver the representation-vectors of $a_1,\ldots,a_m$ resp. $e_1,\ldots,e_n$ in the new coordinate system with respect to the basis $a_{\Delta^1},\ldots,a_{\Delta^n}$.
In addition, we get useful information by having the so-called slacks

$$\begin{aligned} \beta^j &= 1-\alpha_j^{\,T}(A_\Delta^T)^{-1}\mathbb{1} = 1-\alpha_j^{\,T}\,\mathbb{1}, \\ -x^i &= 0-e_i^{\,T}(A_\Delta^T)^{-1}\mathbb{1} = 0-\gamma_i^{\,T}\,\mathbb{1}\,. \end{aligned} \tag{2.2}$$

Here $x = (x^1,\ldots,x^n)^T$ is the normal vector to $H(a_{\Delta^1},\ldots,a_{\Delta^n})$, which satisfies

(2.3) $\quad a_{\Delta^1}{}^T x = 1, \ldots, a_{\Delta^n}{}^T x = 1$.

These data can be stored in the Simplex-Tableau belonging to the boundary simplex $CH(a_{\Delta^1}, \ldots, a_{\Delta^n},)$, which looks as follows:

$$
\begin{array}{c|ccccc|ccc|}
 & a_1 & & a_j & & a_m & e_1 & & e_n \\
\hline
a_{\Delta^1} & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\
\vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\
a_{\Delta^i} & \alpha_1 & & \alpha_j & & \alpha_m & \gamma_1 & (A_\Delta)^{-1} & \gamma_n \\
\vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\
a_{\Delta^n} & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\
\hline
 & \beta^1 & \cdots & \beta^j & \cdots & \beta^m & -x^1 & \cdots & -x^n \\
\hline
\end{array}
$$

<u>Table I</u>

If $a_{\Delta^i} = a_{i_1}$ is to be replaced in the basis, then we have to find a substitute a_{i_2} with $i_2 \notin \Delta$. Using a_{i_2} we obtain a hyperplane $H\left(a_{\Delta^1}, \ldots, a_{\Delta^{i-1}}, a_{i_2}, a_{\Delta^{i+1}}, \ldots, a_{\Delta^n}\right)$, whose normal vector x satisfies

(2.4) $\quad a_i^T x \leq 1 \quad \forall i = 1, \ldots, m \quad$ or $\quad a_i^T x \geq 1 \quad \forall i = 1, \ldots, m$.

That means, that in the new tableau it is required that

(2.5) $\quad \beta^j \geq 0 \; \forall j = 1, \ldots, m \quad$ or $\quad \boxed{} \quad \forall j = 1, \ldots, m$.

The Simplex-Method recognizes the vector a_{i_2} as the extremal argument among the $m–n$ candidates for the so-called quotient-criterion. The geometrical task in determining a_{i_2} is simply to find that point a_j $(j$ ” $\Delta)$ that realizes (by augmentation to $a_{\Delta^1}, \ldots, a_{\Delta^{i-1}}, a_{\Delta^{i+1}}, \ldots, a_{\Delta^n})$ the smallest rotation angle with respect to $CH\left(a_{\Delta^1}, \ldots, a_{\Delta^{i-1}}, a_{i_1}, a_{\Delta^{i+1}}, \ldots, a_{\Delta^n}\right)$.

Depending on the configuration before the pivot step, one has to decide according to the following rules. Let us use $\hat{\beta}^j$ for denoting the slack after the pivot operation (basis exchange).

In general the formula for updating the slacks is $\hat{\beta}^j = \beta^j - \dfrac{\beta^t \cdot \alpha_j^i}{\alpha_t^i}$ when $a_{\Delta^i} = a_{i_1}$ is substituted by a_t.

i) If $\beta \geq 0$ completely and if there are entries $\alpha_j^i < 0$, then determine a t with $\alpha_t^i < 0$, such that $\dfrac{\beta^t}{\alpha_t^i} \geq \dfrac{\beta^j}{\alpha_j^i}$ for all j with $\alpha_j^i < 0$.

Then we have

$\hat{\beta}^j = \beta^j - \dfrac{\beta^t \alpha_j^i}{\alpha_t^i} \geq 0$ for those j's, and for the j's with $\alpha_j^i \geq 0$ anyway.

ii) If $\beta \geq 0$ and if there are no entries $\alpha_j^i < 0$, then determine a t with $\alpha_t^i > 0$, such that $\dfrac{\beta^t}{\alpha_t^i} \geq \dfrac{\beta^j}{\alpha_j^i}$ for all j with $\alpha_j^i > 0$.

Then we have

$\hat{\beta}^j = \beta^j - \dfrac{\beta^t \alpha_j^i}{\alpha_t^i} \leq 0$ for all those j's, and for the j's with $\alpha_j^i \leq 0$ anyway.

iii) If $\beta \leq 0$ and if there are entries $\alpha_j^i < 0$, then determine a t with $\alpha_t^i < 0$, such that $\dfrac{\beta^t}{\alpha_t^i} \leq \dfrac{\beta^j}{\alpha^j}$ for all j with $\alpha_j^i < 0$.

Then we have

$\hat{\beta}^j = \beta^j - \dfrac{\beta^t \alpha_j^i}{\alpha_t^i} \leq 0$ for all those j's, and for the j's with $\alpha_j^i \geq 0$ anyway.

iv) If $\beta \leq 0$ and if there are no entries $\alpha_j^i < 0$, then determine a t with $\alpha_t^i > 0$, such that $\dfrac{\beta^t}{\alpha_t^i} < \dfrac{\beta^j}{\alpha_j^i}$ for all j with $\alpha_j^i > 0$.

Then we have

$\hat{\beta}^j = \beta^j - \dfrac{\beta^t \alpha_j^i}{\alpha_t^i} \geq 0$ for all those j's, and for the j's with $\alpha_j^i \leq 0$ anyway.

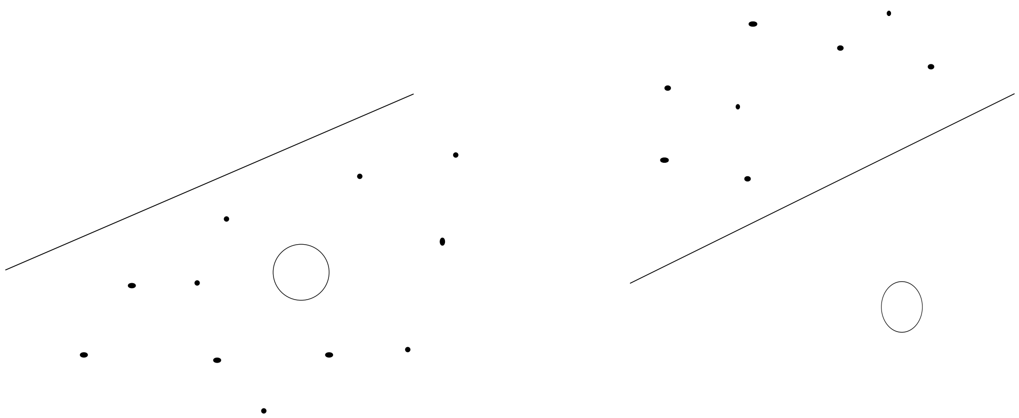
Definition 2.1

In the case of $\beta^1,\ldots,\beta^m \geq 0$, i.e. $a_1^T x \,\%1, \ldots, a_m^T x \,\%1$, we call $CH\left(a_{\Delta^1},\ldots,a_{\Delta^n}\right)$ a facet or boundary simplex of the first kind.
In the case of $\beta^1,\ldots,\beta^m \leq 0$, i.e. $a_1^T x \wedge 1, \ldots, a_m^T x \wedge 1$, we call $CH\left(a_{\Delta^1},\ldots,a_{\Delta^n}\right)$ a facet or boundary simplex of the second kind.
F partitions into F_1 and F_2, which are the sets of facets of first and second kind.

Remark

A boundary simplex of first kind has the property that its affine hull (the hyperplane through the generating points) assembles all remaining points and the origin in the same halfspace.
A boundary simplex of second kind has the property that its affine hull separates the origin from all other remaining points.

Figure 1

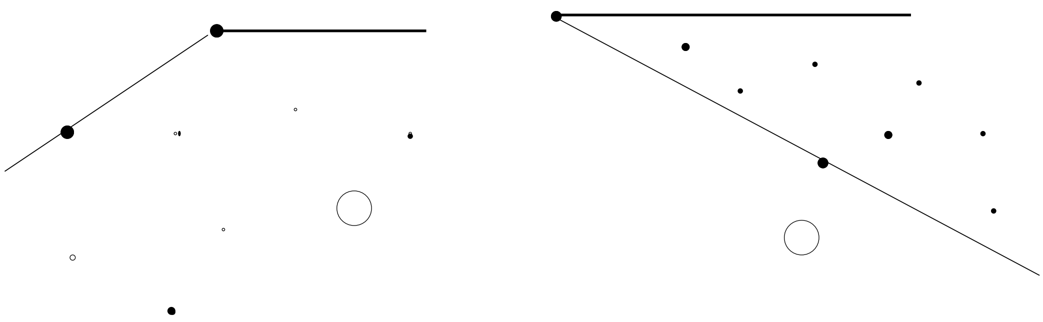


1st kind 2nd kind

Remark

Application of the 4-case quotient-criterion mentioned above corresponds to minimizing the rotation-angle around the axis $AH\left(a_{\Delta^1},\ldots,a_{\Delta^{i-1}},a_{\Delta^{i+1}},\ldots,a_{\Delta^n}\right)$ (AH = affine hull). Starting from the hyperplane $H\left(a_{\Delta^1},\ldots,a_{\Delta^{i-1}},a_{\Delta^i},a_{\Delta^{i+1}},\ldots,a_{\Delta^n}\right)$ we rotate so far, until we meet the first point replacing a_{Δ^i} (called a_{i_2}). This can also be interpreted as cracking the surface at the rotation axis until we get another supporting point. The quotient-criterion delivers that first supporting point.

Figure 2



1st kind 2nd kind

Remark

For an efficient implementation it is only required to update A_Δ^{-1} and x permanently and to generate the tableau-row for α_{Δ^i} and the β-row.
This can be done with one scalar product per entry.
The total effort for the quotient-criterion of one pivot step is therefore $O(mn)$.

3. A simple Gift-Wrapping-Algorithm

We present an algorithm, which discovers all facets.

Definition 3.1

A border of a facet or boundary simplex is called saturated if both augmentation points a_{i_1} and a_{i_2} are known and hence the two incident facets are discovered. It is called unsaturated, if only one augmentation point is known, the other unknown. That means that we did not know and visit the other facet so far.

Definition 3.2 Simple Gift-Wrapping

Initialization

a) Determine a start-facet as follows
   i) Augment the set $\{a_1,...,a_m\}$ by $\rho e_1,...,\rho e_n$ $(\rho > 0)$ with $\rho > \|a_i\|_1$ for all i=1,...,m.
   Now $CH(\rho e_1,...,\rho e_n)$ is a facet of $CH(a_1,...,a_m,\rho e_1,...,\rho e_n)$.
   ii) Eliminate successively the vectors ρe_i from the basis by pivot steps and by entering respective facets of the rest-system.
   iii) After n such exchange-steps (eliminated ρe_i's are not available for entering), we have a start-boundary simplex of the original system in hand.

b) Store the obtained facet in a tree-storing file.

c) Note all borders of the given facet in a lexicographically organized tree-file together with the corresponding augmentation point in a file for unsaturated facet-borders.

Typical Step

1) Search for an unsaturated BSB to the current facet. If there is one, go directly to 4).

2) Carry out reverse pivot steps according to the stored facets in the tree-file until you have a facet with at least one unsaturated BSB.
   If there is one, go to 4).

3) Stop, since the unsaturated facet-borders-file is empty.

4) Leave the current facet across the unsaturated border after determination of the second augmentation point a_{i_2}, i.e. carry out a forward pivot step to discover a new facet. Note the new facet in the facet-file.

5) Determine all BSB's to the current facet .
   For each of them do the following
   i) Check whether this BSB is already in the unsaturated facet-borders-file.
   ii) If yes, then erase the BSB, because it is now saturated, since we have already seen it from the second incident facet.
   iii) If no, then store the BSB in the unsaturated facet-borders-file, since we have seen it the first time.

6) Go to 1).

Theorem 2

The Gift-Wrapping-Algorithm determines and discovers all facets.

Proof
Each forward pivot step enters a new facet, since we are traversing an unsaturated boundary-simplex-border (BSB). The number of facets is bounded by $\binom{m}{n}$, hence the number of forward pivot steps is finite. The algorithm stops only if no unsaturated BSB is known. Then there cannot be an unvisited facet. This is because the facet-set seen as a graph is connected. If there would be a nonvisited facet, then we would have

a path for a pivot-step-walk from the current facet to that unknown facet. On this path at least one unsaturated facet-border must be traversed. This contradicts the potential existence of an unknown facet when the algorithm stops.

Theorem 3

The simple Gift-Wrapping-Algorithm requires an arithmetic effort of not more than

$$C\,\{(\#\mathrm{F} + n)[\,nm + n^2 \ln m]\},$$

where C is a constant.

Proof

In the initialization there are n and in the main process there are #F forward pivot steps without any variation. The only interesting question is the work which has to be done in the single pivot step.

Each updating of A_Δ^{-1} requires n^2 arithmetic operations. At most m-n quotients have to be calculated, for each one two entries have to be reproduced, which costs $2{\cdot}2n$ arithmetic operations.

Searching for an observed BSB in the tree of unsaturated BSB's can be done in $O(n \ln m)$ as well as inserting and/or erasing these BSB's. This relies on the fact that the whole tree cannot have more than $\binom{m}{n}$ leaves.

The number of backtracking pivot steps is less than the number of forward pivot steps, because they lead backward from a dead end to a facet visited before with unsaturated border. This happens exactly on a subpath which had been used before in forward direction. In addition, a backward step will never be repeated, since leaving a facet without unsaturated borders backward means that we never will visit this facet again.

The consequence of these arguments is an upper bound of

$$O\,\{(\#\mathrm{F} + n)[\,nm + n^2 \ln m]\}.$$

Some remarks on data-storing

We store all the unsaturated BSB's (each one with its additional augmentation point) in a search tree in the following way. $\left(a_{\Delta^1},\ldots,a_{\Delta^{i-1}},a_{\Delta^{i+1}},\ldots,a_{\Delta^n}\,\middle|\,a_{i_1}\right)$ may be the information which is to be saved in a pointer- or path-form.

In our tree we find the following nodes for our purpose:

(0) in the highest hierarchical rank	0
(a_{Δ^1}) in rank	1
$(a_{\Delta^1}, a_{\Delta^2})$ in rank	2
$\vdots$	$\vdots$
$(a_{\Delta^1},\dots,a_{\Delta^l})$ in rank	l $(l < i)$
$\vdots$	$\vdots$
$(a_{\Delta^1},\dots,a_{\Delta^{i-1}})$ in rank	$i-1$
$(a_{\Delta^1},\dots,a_{\Delta^{i-1}},a_{\Delta^{i+1}})$ in rank	i
$\vdots$	$\vdots$
$(a_{\Delta^1},\dots,a_{\Delta^{i-1}},a_{\Delta^{i+1}},\dots,a_{\Delta^l})$ in rank	$l-1$ $(l > i)$
$\vdots$	$\vdots$
$(a_{\Delta^1},\dots,a_{\Delta^{i-1}},a_{\Delta^{i+1}},\dots,a_{\Delta^n})$ in rank	$n-1$
$(a_{\Delta^1},\dots,a_{\Delta^{i-1}},a_{\Delta^{i+1}},\dots,a_{\Delta^n} \mid a_{i_1})$ in rank	n

To find a leave of that tree, one has to decide between (up to) m candidates in each stage (step from one rank to another). Note that always $\Delta^1 < \Delta^2 < \dots < \Delta^n$.
In an efficient implementation this single decision can also be organized in a binary tree-structure. So one of the above decisions can be replaced by $\ln m$ binary decisions. Therefore the whole search requires $O(n \ln m)$ decisions. The same holds for insertion and for erasure of paths.

4. Accelerating the Gift-Wrapping Algorithm

In the general case not all m points may be vertices of Y. In this case they "disappear" in the interior of Y. We know that

(4.1) $\quad \mathrm{V} = \{a_i \mid a_i \text{ vertex of } CH(a_1,...,a_m)\} \subset \{a_1,...,a_m\}$, but
$\quad CH(\mathrm{V}) = CH(a_1,...,a_m)$.

If already a subset of the points is capable to generate the complete convex hull, then saving effects seem possible. It would be fine if we could replace the factor m by a lower number, hopefully by the number of vertices #V. But how should we realize the appropriate algorithm and how should we distinguish vertices from other points in advance?

Let us briefly remember the distinction between facets of first and of second kind. A facet is of first kind, if its affine hull bounds one halfspace containing the origin and all remaining points. It is of second kind, if the corresponding halfspace again contains all points, but not the origin.

Now we present a concept for savings while determining the more frequently occurring facets of first kind. Before, we should introduce some necessary notation.

Definition 4.1

Corresponding to a simplex $CH(a_{\Delta^1},...,a_{\Delta^n})$, define $H(a_{\Delta^1},...,a_{\Delta^n})$ as the hyperplane containing the complete simplex and define $h(a_{\Delta^1},...,a_{\Delta^n})$ as its height, i.e. the distance between origin and $H(a_{\Delta^1},...,a_{\Delta^n})$.

The following auxiliary result helps to exclude certain points from participation in the process of determining a certain facet.

Lemma 4.1

a) If $\|a_{\Delta^i}\| < \bar{h}$ for a simplex $CH(a_{\Delta^1},...,a_{\Delta^i},...,a_{\Delta^n})$, then $h(a_{\Delta^1},...,a_{\Delta^n}) < \bar{h}$.

b) All simplices with $h\left(a_{\Delta^1},...,a_{\Delta^n}\right) \geq \bar{h}$ are generated only by points a_{Δ^i} such that $\left\|a_{\Delta^i}\right\| \geq \bar{h}$.

c) Let $CH\left(a_{\Delta^1},...,a_{\Delta^n}\right)$ be a boundary simplex of first kind for $CH(a_1,...,a_l)$ $(l < m)$ and let $\left\|a_{l+1}\right\|,...,\left\|a_m\right\| < h\left(a_{\Delta^1},...,a_{\Delta^n}\right)$. Then $CH\left(a_{\Delta^1},...,a_{\Delta^n}\right)$ is a boundary simplex of $CH(a_1,...,a_m)$ as well.

d) If the quotient-criterion predicts a new boundary simplex of first kind $CH\left(a_{\Delta^1},...,a_{\Delta^n}\right)$ after evaluation of the points $a_1,...,a_l$, and if simultaneously $\left\|a_{l+1}\right\|,...,\left\|a_m\right\| \leq h\left(a_{\Delta^1},...,a_{\Delta^n}\right)$, the prediction is correct, and $a_{l+1},...,a_m$ need not be checked further.

Proof

a) and b) are opposite formulations of the same insight, which is based on the fact that $\left\|a_{\Delta^i}\right\| \geq h\left(a_{\Delta^1},...,a_{\Delta^n}\right)$ for all $i=1,...,n$.

c) holds, because the points $a_{l+1},...,a_m$ cannot question the property of being a boundary simplex.

d) is simply the algorithmic consequence from *c)*.

Motivated by that lemma we prefer to sort the points according to their Euclidean norm (in decreasing order). After that we apply the quotient-criterion index-increasing resp. length-decreasing. As soon as the length of the vector under consideration becomes smaller than the height of the existing boundary simplex, we can stop, because the boundary simplex is fixed now. The remaining quotient-calculations become superfluous and useless. Since it may not be necessary to sort all points, we can (for further acceleration) implement a heap for dynamical sorting according to the norm. We sort only as far as necessary.

Definition 4.2

We denote the set of already sorted points by S. These are the #S vectors of greatest Euclidean length. For a certain height-value h, denote by $s(h)$ the number of points satisfying $\left\|a_i\right\| \geq h$.

Now we are able to formulate a speed-up-version of our algorithm.

Definition 4.3 Accelerated Gift-Wrapping Algorithm

Initialization

a) Implement a heap for sorting points dynamically according to their length in decreasing order.

b) Determine a start-facet as follows (as before in Def. 3.2)
   i) Augment the set $\{a_1,...,a_m\}$ by $\rho e_1,...,\rho e_n$ $(\rho > 0)$ with $\rho > \|a_i\|_1$ for all i=1,...,m.
   Now $CH(\rho e_1,...,\rho e_n)$ is a facet of $CH(a_1,...,a_m,\rho e_1,...,\rho e_n)$.
   ii) Eliminate successively the vectors ρe_i from the basis by pivot steps and by entering respective facets of the rest-system.
   iii) After n such exchange-steps (eliminated ρe_i's are not available for entering), we have a start-boundary-simplex of the original system at hand.

c) Store the obtained facet in a tree-storing file.

d) Note all borders of the given facet in a lexicographically organized tree-file together with the corresponding augmentation point in a file for unsaturated facet borders.

Typical step

1) Search for an unsaturated BSB to the current facet. If there is one, go directly to 4).

2) Carry out reverse pivot steps according to the stored facets in the tree-file until you have a facet with at least one unsaturated BSB.
   If there is one, go to 4).

3) Stop, since the unsaturated facet-borders-file is empty.

4) Leave the current facet across the unsaturated border after determination of the second augmentation point a_{i_2}. For that purpose do the following for $j=1,...,m$ starting from $j=1$.
   i) Determine the j-longest vector (new a_j) either from the list of sorted points or by using the heap; repeat with $j=j+1$ if a_j [Δ.
   ii) Apply the quotient-criterion and determine the current optimum and the preliminary substitute $\tilde{a}_{i_2}$.
   iii) If $CH\left(a_{\Delta^1},...,a_{\Delta^{i-1}},\tilde{a}_{i_2},a_{\Delta^{i+1}},...,a_{\Delta^n}\right)$ is of second kind, then set $j=j+1$ and go to i).
   iv) If $CH\left(a_{\Delta^1},...,a_{\Delta^{i-1}},\tilde{a}_{i_2},a_{\Delta^{i+1}},...,a_{\Delta^n}\right)$ is of first kind, then check whether $h\left(a_{\Delta^1},...,a_{\Delta^{i-1}},\tilde{a}_{i_2},a_{\Delta^{i+1}},...,a_{\Delta^n}\right) \leq \left\|a_j\right\|$. If yes and $j<m$, set $j=j+1$ and go to i).

   Note the new facet in the facet-file.

5) To the current facet determine all BSB's. For each of them do the following:
   i) Check whether this BSB is already in the unsaturated borders-file.
   ii) If yes, then erase the BSB, because it is saturated now, since we have seen it from the second facet.
   iii) If no, then store the BSB in the unsaturated facet-borders-file, since we have seen it the first time.

6. Go to 1).

<u>Theorem 4</u>

The accelerated algorithm discovers all facets.

<u>Proof</u>

Essentially, we have changed only 4). Here we refer to lemma 4.1. This allows us to dispense with calculating quotients as soon as we have a preliminary facet of certain height h and as all following points satisfy $\left\|a_j\right\| \leq h$.

The sorting process is continued if necessary in 4i).

Theorem 5

The total effort for the accelerated algorithm is

$$O\{n[n^2 + mn + n^2 \ln m] + \sum_{BS\in F_1} [n^2 + s(h_{BS})n + n^2 \ln m] + \sum_{BS\in F_2} [n^2 + mn + n^2 \ln m] + m + (\ln m)\, s(Min^{\geq 0}\, h_F)\} \,.$$

Here h_{BS} is the actual height of the wanted facet (BS) and $s(h_{BS})$ is the number of points with greater Euclidean length. In addition we use

$$Min^{\geq 0}\, h_{BS} := \begin{cases} \underset{BS\in F_1}{Min}\, h_{BS} & \text{if } F_2 = \emptyset \\ 0 & \text{if } F_2 \neq \emptyset \end{cases}$$

Proof

The proof is as before, but in the situation of a boundary simplex of first kind we need only $s(h_{BS})$ instead of m quotients. New is also the requirement to calculate the height of the preliminary boundary simplex. But this can also be done in $O(n)$ time per point such that the order mentioned above is not affected.

$O(m)$-time is now needed for implementation of the heap and also the last term (for sorting as many points as required) is new.

5. A probabilistic model

Now we present a family of distributions which shall serve as a basis for the probabilistic considerations.

Definition 5.1

Let $a_1,\ldots,a_m$ be distributed symmetrically under rotations, identically and independently over the n-dimensional unit ball (without origin).

Remark

Such a distribution-model for our points satisfies almost surely our condition of nondegeneracy (1.1).

Definition 5.2

A rotation-symmetric distribution can be characterized uniquely by its "radial distribution function".

$$F(r) := P(\|x\| \le r) \quad \text{for } r \in [0, \infty),$$

where $P(\|x\| \le r)$ denotes the probability, that a random point x is of Euclidean length not greater than r.

In our family of distributions over the unit ball it is clear, that always $F(r)=1$ for $r \ge 1$.

If the radial distribution function has a density, then we denote it by $f(r)$, i.e.

$$(5.1) \qquad F(r) = \int_0^r f(\rho)\, d\rho \qquad \text{for all } r \in [0, \infty).$$

If the distribution over R^n has a density $\hat{f}$, then we have

(5.2) $$F(r) := \lambda_{n-1}(\omega_n)\, n \int_0^r t^{n-1} \hat{f}(t)\, dt$$

where $\hat{f}(x) = f(r) \cdot \dfrac{1}{n \cdot r^{n-1} \cdot \lambda_{n-1}(\omega_n)}$.

Here $\hat{f}$ is a density over R^n, hence $\hat{f} : R^n \to R^+$

such that $\hat{f}(x_1) = \hat{f}(x_2)$ for all x_1, x_2 with $\|x_1\| = \|x_2\|$.

Definition 5.3 (notation for balls)

ω_n denotes the n-dimensional unit sphere $\{x \in R^n \mid \|x\| = 1\}$
and Ω_n the n-dimensional unit ball $\{x \in R^n \mid \|x\| \leq 1\}$.
$\lambda_k(\cdot)$ stands for the k-dimensional Lebesgue-measure of the specified set, hence

(5.3) $$\lambda_{n-1}(\omega_n) = \frac{2\pi^{n/2}}{\Gamma\left(\frac{n}{2}\right)} \text{ and } \lambda_n(\Omega_n) = \frac{\pi^{n/2}}{\Gamma\left(\frac{n+2}{2}\right)} .$$

In this paper we restrict to a special class of distributions over Ω_n, which is particularly suitable for the necessary integration-operations and – on the other hand – demonstrates all typical features of changing the weight or mass of the distribution from the boundary (sphere) of the ball to the center (origin) of the ball.

Definition 5.4

Under variation of a parameter $k \in (-1, \infty)$ we define

$$F_k(r) := \begin{cases} \dfrac{\int_0^r (1-\tau^2)^k \tau^{n-1} d\tau}{\int_0^1 (1-\tau^2)^k \tau^{n-1} d\tau} & \text{for } 0 \leq r \leq 1 \\ 1 & \text{for } r > 1 \end{cases} .$$

These radial distributions have corresponding radial densities, namely

(5.4) $$f_k(r) := \frac{(1-r^2)^k\, r^{n-1}}{\int_0^1 (1-\tau^2)^k \tau^{n-1} d\tau} \qquad \text{for } \ 0 \le r \le 1 \quad \text{and} \ 0 \ \text{elsewhere.}$$

<u>Remark</u>

The parameter k gives the weighting on the radii between 0 and 1. $k \to -1$ means extremal weight at the boundary of the ball, and increasing k describes an increment of weight in the interior and a decrease at the sphere.

Exploiting the relation (5.2) we obtain

(5.5) $$f(r) \ = \ \lambda_{n-1}(\omega_n)\, n\, r^{n-1}\, \hat{f}\,(x) \qquad \text{for all } x \text{ with } \|x\| = r$$

and for our special case with k

(5.6) $$\hat{f}\,(x) \ = \ \frac{f(\|x\|)}{\lambda_{n-1}(\omega_n)\, n\, \|x\|^{n-1}} \ = \ \frac{\left(1-\|x\|^2\right)^k}{\lambda_{n-1}(\omega_n)\, n \int_0^1 (1-\tau^2)^k\, \tau^{n-1} d\tau} \ .$$

<u>Remark</u>

Interesting special cases are:

(5.7) k=0 □ $\hat{f}$ constant on Ω_n $\hat{=}$ uniform distribution on Ω_n.

(5.8) $k \to -1$ □ extremal dominance at r=1 $\hat{=}$ uniform distribution on ω_n.

(5.9) $k \to \infty$ □ extremal dominance at r=0 $\hat{=}$ totally centralized.

(5.10) $k = \frac{n-1}{2}$ □ radial density symmetric around $r = \frac{1}{2}$.

Equipped with these tools we are going to calculate the necessary expected values corresponding to theorem 5. But before, let us specialize the formulas mentioned above to our special cases.
We use the following substitution rule.

Lemma 5.1

a) $\int_0^1 (1-\tau^2)^k \tau^{n-1} d\tau \;=\; \int_0^1 (1-\tau^2)^k \tau \cdot \tau^{n-2} d\tau =$ (by substitution $u = \tau^2$)

$$= \frac{1}{2}\int_0^1 (1-u)^k u^{\frac{n-2}{2}} du \;=\; \frac{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}{2\Gamma\left(k+1+\frac{n}{2}\right)} = \frac{1}{2} B\left[k+1, \frac{n}{2}\right]$$

where $\Gamma(\cdot)$ and $B(\cdot,\cdot)$ denote the well known Gamma- and Betafunction.

b) $\int_h^1 (r^2 - h^2)^l (1-r^2)^k r\, dr \;=$ (by substitution $u = \frac{r^2 - h^2}{1-h^2}$)

$$= \int_0^1 u^l (1-h^2)^l (1-u)^k (1-h^2)^k (1-h^2) \cdot \frac{1}{2} du \;=\; \frac{1}{2}(1-h^2)^{l+k+1} \int_0^1 u^l (1-u)^k \; du \;=$$

$$= \frac{1}{2}(1-h^2)^{l+k+1} \frac{\Gamma(k+1)\Gamma(l+1)}{\Gamma(k+l+2)} \;=\; \frac{1}{2}(1-h^2)^{l+k+1} B(k+1, l+1).$$

More detailed explanations concerning technical details can be obtained from [Borgwardt (1987)] Appendix 6.1.

Now we formulate our special versions of functions with parameter k, which appears as subscript resp. second subscript index.

Definition 5.5

For these distributions we can also study the marginal distribution and the marginal density.
The marginal distribution function is defined by

$$G : [-1, 1] \to [0, 1] \quad \text{and} \quad G(h) := P(x^n \leq h).$$

The marginal density function is defined by

$$g : [-1, 1] \varnothing [0, `] \quad \text{and} \quad \int_{-1}^{h} g(\mathrm{h})\, d\mathrm{h} \;=\; G(h).$$

For general rotation symmetric distributions the following marginal distribution- and related functions are useful for our purposes.

(5.11) $$G(h) \;=\; 1-\frac{\lambda_{n-2}(\omega_{n-1})}{\lambda_{n-1}(\omega_n)}\int_h^1\int_{\frac{h}{r}}^1 (1-\sigma^2)^{\frac{n-3}{2}}\, d\sigma\, dF(r)$$

(5.12) $$g_0(h) := g(h) \;=\; \frac{\lambda_{n-2}(\omega_{n-1})}{\lambda_{n-1}(\omega_n)}\int_h^1 \frac{(r^2-h^2)^{\frac{n-3}{2}}}{r^{n-2}}\, dF(r)$$

(5.13) $$g_1(h) := g_0(h)\cdot E\left(\left|x^1\right| / x^n = h\right) \;=$$

$$=\; \frac{2\lambda_{n-3}(\omega_{n-2})}{(n-2)\lambda_{n-1}(\omega_n)}\int_h^1 \frac{(r^2-h^2)^{\frac{n-2}{2}}}{r^{n-2}}\, dF(r) \;=\; \frac{1}{\pi}\int_h^1 \frac{(r^2-h^2)^{\frac{n-2}{2}}}{r^{n-2}}\, dF(r)\,.$$

(5.14) $$g_2(h) := g_0(h)\cdot E\left(\left(x^1\right)^2 / x^n = h\right) \;=$$

$$=\; \frac{\lambda_{n-2}(\omega_{n-1})}{(n-1)\lambda_{n-1}(\omega_n)}\int_h^1 \frac{(r^2-h^2)^{\frac{n-1}{2}}}{r^{n-2}}\, dF(r)\,.$$

Then (5.11 - 5.15) specialize to

(5.15) $$G_k(h) \;=\; 1-\frac{\lambda_{n-2}(\omega_{n-1})}{\lambda_{n-1}(\omega_n)}\int_h^1\int_{\frac{h}{r}}^1 \left(1-\sigma^2\right)^{\frac{n-3}{2}}\, d\sigma\,(1-r^2)^k r^{n-1}\, dr \;\cdot\; \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}$$

(5.16) $$g_{0,k}(h) \;=\; \frac{\lambda_{n-2}(\omega_{n-1})}{\lambda_{n-1}(\omega_n)}\cdot\frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}\int_h^1 \frac{(r^2-h^2)^{\frac{n-3}{2}}}{r^{n-2}}\,(1-r^2)^k r^{n-1}\, dr \;=$$

$$=\; \frac{\lambda_{n-2}(\omega_{n-1})\,2\Gamma\left(k+1+\frac{n}{2}\right)}{\lambda_{n-1}(\omega_n)\,\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}\cdot\int_h^1 (r^2-h^2)^{\frac{n-3}{2}}\,(1-r^2)^k\, r\, dr \;=$$

$$= \frac{\lambda_{n-2}(\omega_{n-1})\, 2\Gamma\left(k+1+\frac{n}{2}\right)}{\lambda_{n-1}(\omega_n)\, \Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \cdot \frac{1}{2}(1-h^2)^{\frac{n-3}{2}+k+1} \frac{\Gamma\left(\frac{n-1}{2}\right)\Gamma(k+1)}{\Gamma\left(\frac{n-1}{2}+k+1\right)} =$$

$$= \frac{\lambda_{n-2}(\omega_{n-1})\, \Gamma\left(\frac{n-1}{2}\right)\Gamma\left(k+1+\frac{n}{2}\right)}{\lambda_{n-1}(\omega_n)\, \Gamma\left(\frac{n}{2}\right)\Gamma\left(k+1+\frac{n-1}{2}\right)} (1-h^2)^{\frac{n-1}{2}+k} =$$

$$= \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\sqrt{\pi}\;\Gamma\left(k+1+\frac{n-1}{2}\right)} \cdot (1-h^2)^{\frac{n-1}{2}+k}.$$

$$(5.17)\quad g_{1,k}(h) = \frac{1}{\pi}\int_h^1 \frac{(r^2-h^2)^{\frac{n-2}{2}}}{r^{n-2}} (1-r^2)^k\, r^{n-1}\, dr \cdot \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} =$$

$$= \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\pi\cdot\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \int_h^1 (r^2-h^2)^{\frac{n-2}{2}} (1-r^2)^k\, r\, dr =$$

$$= \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\pi\cdot\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \cdot \frac{1}{2}(1-h^2)^{\frac{n}{2}+k} \frac{\Gamma\left(\frac{n}{2}\right)\Gamma(k+1)}{\Gamma\left(k+1+\frac{n}{2}\right)} =$$

$$= \frac{1}{\pi}(1-h^2)^{\frac{n}{2}+k}.$$

$$(5.18)\quad g_{2,k}(h) = \frac{\lambda_{n-2}(\omega_{n-1})}{(n-1)\lambda_{n-1}(\omega_n)} \int_h^1 \frac{(r^2-h^2)^{\frac{n-1}{2}}}{r^{n-2}} (1-r^2)^k\, r^{n-1}\, dr \cdot \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} =$$

$$= \frac{\lambda_{n-2}(\omega_{n-1})}{(n-1)\lambda_{n-1}(\omega_n)} \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \cdot \int_h^1 (r^2-h^2)^{\frac{n-1}{2}} (1-r^2)^k\, r\, dr =$$

$$= \frac{\lambda_{n-2}(\omega_{n-1})}{(n-1)\lambda_{n-1}(\omega_n)} \; \frac{2\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \; \frac{1}{2}(1-h^2)^{\frac{n+1}{2}+k} \; \frac{\Gamma\left(\frac{n+1}{2}\right)\Gamma(k+1)}{\Gamma\left(\frac{n+1}{2}+k+1\right)} \; =$$

$$= \frac{\lambda_{n-2}(\omega_{n-1})}{(n-1)\lambda_{n-1}(\omega_n)} \; \frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} \; \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma\left(\frac{n+1}{2}+k+1\right)} \; (1-h^2)^{\frac{n+1}{2}+k} \; =$$

$$= \frac{1}{\sqrt{\pi}} \frac{\Gamma\left(\frac{n}{2}\right)\Gamma\left(\frac{n+1}{2}\right)}{(n-1)\,\Gamma\left(\frac{n-1}{2}\right)\Gamma\left(\frac{n}{2}\right)} \; \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma\left(k+1+\frac{n+1}{2}\right)} \; (1-h^2)^{\frac{n+1}{2}+k} \; =$$

$$= \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\;\Gamma\left(k+1+\frac{n+1}{2}\right)} \; (1-h^2)^{\frac{n+1}{2}+k} \; .$$

6. The expected complexity for our algorithm

We remind of our upper bound for the total effort (theorem 5)

$$O\{n[n^2 + mn + n^2 \ln m] + \sum_{BS \in F_1} [n^2 + s(h_{BS})n + n^2 \ln m] +$$

$$+ \sum_{BS \in F_2} [n^2 + mn + n^2 \ln m] + m + (\ln m)\, s(Min^{\geq 0}\, h_F)\} \;.$$

For achieving an upper bound for the expected total effort, we need the expected value of the bound mentioned above..

Since constant dimension factors can be treated trivially, only the following expectation values are interesting:

$$E^k_{m,n}(\#F_1),\; E^k_{m,n}(\#F_2),\; E^k_{m,n}\Big(\sum_{BS \in F_1} s(h_{BS})\Big),\; E^k_{m,n}(Min^{\geq 0}\, h_F).$$

Here $E^k_{m,n}$ stands for the expected value for m points, dimension n and parameter k. These expectation values will now be calculated. We rely on the methods and techniques developed in [Borgwardt (1987)] for dealing with these types of integrals.

In general we will exploit the linearity of expectation values. That means we count the potential candidates for having a certain property and multiply this number with the probability that a typical candidate actually satisfies the conditions. Here we can make use of the assumptions on identical and independent distributions. This general approach delivers the expected total numbers.

We start with $\#F_1$. There are $\binom{m}{n}$ n-tuples $(a_{\Delta^1},\ldots,a_{\Delta^n})$ and the probability can be derived by integrating over all configurations of n points.

From [Borgwardt (1987)] we know that

$$(6.1)\quad E^k_{m,n}(\#F_1) = \binom{m}{n} \int_{\mathbb{R}^n}\cdots\int_{\mathbb{R}^n} G\big(h(a_1,\ldots,a_n)\big)^{m-n}\; d\hat{F}(a_1)\ldots d\hat{F}(a_n) =$$

$$= \binom{m}{n}\, \lambda_{n-1}(\omega_n) \int_0^1 G(h)^{m-n} \int_{\mathbb{R}^{n-1}}\cdots\int_{\mathbb{R}^{n-1}} |\mathrm{Det}\, B|\;\; \hat{f}(b_1)\cdots\hat{f}(b_n)\, d\bar{b}_1\cdots d\bar{b}_n\, dh$$

where

$$B = \begin{bmatrix} b_1^1 & \cdots & b_n^1 \\ \vdots & & \vdots \\ b_1^{n-1} & & b_n^{n-1} \\ 1 & & 1 \end{bmatrix}$$

$$= \binom{m}{n} \lambda_{n-1}(\omega_n) \int_0^1 G(h)^{m-n} \quad \Lambda_1(h)\, dh,$$

with $\Lambda_1(h) := \int_{\mathbb{R}^{n-1}} \cdots \int_{\mathbb{R}^{n-1}} |\text{Det } B| \;\hat{f}(b_1) \cdots \hat{f}(b_n) \;\; d\bar{b}_1 \cdots d\bar{b}_n$.

The Cauchy-Schwartz-inequality enables us to get a simpler upper bound for $\Lambda_1(h)$ by use of functions $\Lambda_0(h)$ and $\Lambda_2(h)$.

(6.2) $$\Lambda_0(h) := \int_{\mathbb{R}^{n-1}} \cdots \int_{\mathbb{R}^{n-1}} \hat{f}(b_1) \cdots \hat{f}(b_n) \;\; d\bar{b}_1 \cdots d\bar{b}_n$$

(6.3) $$\Lambda_2(h) := \int_{\mathbb{R}^{n-1}} \cdots \int_{\mathbb{R}^{n-1}} |\text{Det } B|^2 \;\hat{f}(b_1) \cdots \hat{f}(b_n) \;\; d\bar{b}_1 \cdots d\bar{b}_n .$$

Then it is clear that

(6.4) $$\Lambda_1(h) \le \left[\Lambda_0(h)\, \Lambda_2(h)\right]^{\frac{1}{2}}$$

$\Lambda_1(h)$ exists, since our distribution has bounded support.

From [Borgwardt (1987)] we know that

(6.5) $$\Lambda_0(h) = g_0(h)^n$$

(6.6) $$\Lambda_2(h) = n!\, g_2(h)^{n-1}\, g_0(h).$$

Consequently

(6.7) $$\Lambda_1(h) \le \left[\Lambda_0(h)\, \Lambda_2(h)\right]^{\frac{1}{2}} = [n!]^{\frac{1}{2}}\, g_0(h)^n \cdot \left[\frac{g_2(h)}{g_0(h)}\right]^{\frac{n-1}{2}} .$$

Specialization delivers according to (5.16) and (5.18)

(6.8) $$\Lambda_{1,k}(h) \le [n!]^{\frac{1}{2}} \; g_{0,k}(h) \cdot g_{0,k}(h)^{\frac{n-1}{2}} \cdot \left[\frac{1}{2(k+1+\frac{n-1}{2})}(1-h^2)\right]^{\frac{n-1}{2}} =$$

$$= [n!]^{\frac{1}{2}} \; g_{0,k}(h) \cdot \left[\frac{1}{\sqrt{\pi}} \frac{\Gamma(k+1+\frac{n}{2})}{\Gamma(k+1+\frac{n-1}{2})}(1-h^2)^{\frac{n-1}{2}+k}\right]^{n-1} \cdot$$

$$\cdot \left[\frac{1}{2(k+1+\frac{n-1}{2})}(1-h^2)\right]^{\frac{n-1}{2}} =$$

$$= [n!]^{\frac{1}{2}} \; g_{0,k}(h) \cdot \left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}} \frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}} \cdot$$

$$\cdot \; (1-h^2)^{\left[\frac{n}{2}+k\right](n-1)}.$$

So we get

(6.9) $$E^k_{m,n}\;(\#F_1) = \binom{m}{n} \; \lambda_{n-1}(\omega_n) \int_0^1 G_k(h)^{m-n} g_{0,k}(h)(1-h^2)^{\left[\frac{n}{2}+k\right](n-1)} dh \; \cdot$$

$$\cdot \; [n!]^{\frac{1}{2}} \left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}} \frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}} .$$

Now there is a relation between $G_k(h)$ and $(1-h^2)$, namely

(6.10) $$1-G_k(h) = \int_h^1 g_{0,k}(\mathrm{h})\, d\mathrm{h} = \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\sqrt{\pi}\;\Gamma\left(k+1+\frac{n-1}{2}\right)} \cdot \int_h^1 (1-h^2)^{\frac{n-1}{2}+k} h \, dh \;\ge$$

$$\geq \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\sqrt{\pi}\;\Gamma\left(k+1+\frac{n-1}{2}\right)} \quad = \int_h^1 (1-h^2)^{\frac{n-1}{2}+k}\, h\, dh \;=$$

$$= \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\sqrt{\pi}\;\Gamma\left(k+1+\frac{n-1}{2}\right)}\;\frac{1}{2}\;\frac{1}{\frac{n+1}{2}+k}\;(1-h^2)^{\frac{n+1}{2}+k}\;.$$

Since the upper bound for $\Lambda_{1,k}(h)$ from (6.8) appearing in (6.9) is essentially a power of $(1-h^2)$, we can substitute it by something like a power of $1-G_k(h)$.

(6.11) $$E^k_{m,n}\;(\#(\mathrm{F}_1)) \;\leq\; \binom{m}{n}\;\lambda_{n-1}(\omega_n)\int_0^1 G_k(h)^{m-n}\;g_{0,k}(h)\;\cdot$$

$$\cdot\;\left(1-G_k(h)\right)^{\frac{\left[\frac{n}{2}+k\right](n-1)}{\frac{n+1}{2}+k}}\;dh\;\cdot$$

$$\cdot\;[n!]^{\frac{1}{2}}\;\left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}}\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}}\;\cdot$$

$$\cdot\left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\sqrt{\pi}\;\Gamma\left(k+1+\frac{n-1}{2}\right)}\;\frac{1}{2}\;\frac{1}{\frac{n+1}{2}+k}\right]^{-\frac{\left[\frac{n}{2}+k\right](n-1)}{\frac{n+1}{2}+k}} =$$

$$= \binom{m}{n}\;\lambda_{n-1}(\omega_n)\;[n!]^{\frac{1}{2}}\;\left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}}\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}}\;\cdot$$

$$\cdot\;\left[\frac{\sqrt{\pi}\;\;\Gamma\left(k+1+\frac{n-1}{2}\right)(n+1+2k)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{n-1-\frac{n-1}{n+1+2k}}\;\cdot$$

$$\cdot\int_0^1 G_k(h)^{m-n}\;g_{k,0}(h)\;\left(1-G_k(h)\right)^{n-1-\frac{n-1}{n+1+2k}}\;dh\;=$$

$$= \binom{m}{n} \lambda_{n-1}(\omega_n)\,[n!]^{\frac{1}{2}} \left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}} \left[\frac{n+1+2k}{k+1+\frac{n-1}{2}}\right]^{\frac{n-1}{2}} \pi^{\frac{n-1}{2}} \;\cdot$$

$$\cdot \left[\frac{2\sqrt{\pi}\;\Gamma\left(k+1+\frac{n+1}{2}\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{-\frac{n-1}{n+1+2k}} \cdot\; (n+1+2k)^{\frac{n-1}{2}} \;\cdot$$

$$\cdot \int_{\frac{1}{2}}^{1} G^{m-n}\,(1-G)^{n-1-\frac{n-1}{n+1+2k}}\,dG \;\le\; \text{(see [Borgwardt (1987)], Appendix)}$$

$$\le \frac{1}{n} \cdot m^{\frac{n-1}{n+1+2k}} \cdot \lambda_{n-1}(\omega_n)\,[n!]^{\frac{1}{2}}\;(n+1+2k)^{\frac{n-1}{2}} \;\cdot$$

$$\cdot \left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\;\Gamma\left(k+1+\frac{n+1}{2}\right)}\right]^{\frac{n-1}{n+1+2k}} =$$

$$= \frac{1}{n} \cdot m^{\frac{n-1}{n+1+2k}}\;\; 2\pi^{\frac{n}{2}}\; \frac{[n!]^{\frac{1}{2}}}{\Gamma\left(\frac{n}{2}\right)}\;(n+1+2k)^{\frac{n-1}{2}} \;\cdot$$

$$\cdot \left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\;\Gamma\left(k+1+\frac{n+1}{2}\right)}\right]^{\frac{n-1}{n+1+2k}} .$$

<u>Remark</u>

The Stirling formula approximates in the following way

$$(6.12)\quad \frac{[n!]^{\frac{1}{2}}}{\Gamma\left(\frac{n}{2}\right)} \;\sim\; \frac{\sqrt{n(n-1)}\;\left(\left[2\left(\frac{n}{2}-1\right)\right]!\right)^{\frac{1}{2}}}{\left(\frac{n}{2}-1\right)^{\frac{n}{2}}\sqrt{2\pi}\;e^{-\frac{n}{2}+1}} \;=$$

$$= \frac{\sqrt{n(n-1)}\ \left[2\left(\frac{n}{2}-1\right)\right]^{\left(\frac{n}{2}-1\right)+\frac{1}{4}}\ e^{\left(\frac{n}{2}-1\right)}}{\left(\frac{n}{2}-1\right)^{\frac{n}{2}}\qquad e^{-\frac{n}{2}+1}} =$$

$$= \sqrt{n(n-1)}\ \ 2^{\left(\frac{n}{2}-1\right)+\frac{1}{4}}\ \left(\frac{n}{2}-1\right)^{-\frac{3}{4}}$$ which is about $n^{\frac{1}{4}} \cdot 2^{\frac{n}{2}}$.

(6.13) The quotient $\dfrac{\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma\left(k+1+\frac{n+1}{2}\right)}$ can be approximated by at least

$\left(k+1+\frac{n}{2}\right)^{-\frac{1}{2}}$ and by at most $\left(k+\frac{n+1}{2}\right)^{-\frac{1}{2}}$.

So the approximation comes up to

(6.14) $E_{m,n}^{k}(\#\mathrm{F}_1) \approx \dfrac{1}{n}\ m^{\frac{n-1}{n+1+2k}}\ 2^{\frac{n}{2}}\ \pi^{\frac{n}{2}} \cdot n^{\frac{1}{4}}\,(n+1+2k)^{\frac{n-1}{2}}\ \cdot$

$\cdot\ \left(k+1+\frac{n}{2}\right)^{-\frac{1}{2}\cdot\frac{n-1}{n+1+2k}}\ \cdot$ Const.

Analoguously we can estimate $E^k_{m,n}(\#\mathrm{F}_2)$. Here we have to study the situation, where all points lie beyond H.

(6.15) $$E^k_{m,n}(\#\mathrm{F}_2) = \binom{m}{n} \lambda_{n-1}(\omega_n) \int_{\mathbb{R}^n}\cdots\int_{\mathbb{R}^n}\left[1-G\big(h(a_1,\ldots,a_n)\big)\right]^{m-n}$$

$$d\hat{F}(a_1)\ldots d\hat{F}(a_n) =$$

$$= \binom{m}{n} \lambda_{n-1}(\omega_n)\int_0^1\left[1-G(h)\right]^{m-n} \Lambda_1(h)\,dh = \text{(in our special case)}$$

$$= \binom{m}{n} \lambda_{n-1}(\omega_n)\int_0^1\left[1-G_k(h)\right]^{m-n} g_{0,k}(h)\left[1-G_k(h)\right]^{\frac{\left[\frac{n}{2}+k\right](n-1)}{\frac{n+1}{2}+k}} dh\ \cdot$$

$$\cdot\ [n!]^{\frac{1}{2}} \left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}} \frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}}\ \cdot$$

$$\cdot \left[\frac{\Gamma(k+1+\frac{n}{2})}{\sqrt{\pi}\ \Gamma(k+1+\frac{n-1}{2})}\ \frac{1}{2}\ \frac{1}{\frac{n+1}{2}+k}\right]^{-\frac{\left[\frac{n}{2}+k\right](n-1)}{\frac{n+1}{2}+k}} =$$

$$= \binom{m}{n} \lambda_{n-1}(\omega_n)\,[n!]^{\frac{1}{2}}\ [n+1+2k]^{\frac{n-1}{2}}\ \cdot$$

$$\cdot \left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}\right]^{\frac{n-1}{n+1+2k}} \cdot \int_{\frac{1}{2}}^{1}(1-G)^{m-n+(n-1)-\frac{n-1}{n+1+2k}}\,dG =$$

$$= \binom{m}{n} \frac{2\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}\right)}\ [n!]^{\frac{1}{2}}\ [n+1+2k]^{\frac{n-1}{2}} \cdot \left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}\right]^{\frac{n-1}{n+1+2k}}\ \cdot$$

$$\cdot\ \frac{1}{m-\frac{n-1}{n+1+2k}}\cdot\frac{1}{2}^{m-\frac{n-1}{n+1+2k}}.$$

This expression tends quickly to 0 when n is fixed and m increases.

<u>Remark</u>

An approximation like in (6.12) and (6.13) delivers

$$(6.16)\quad E_{m,n}^{k}(\#F_2) \approx \binom{m}{n} \frac{1}{m-\frac{n-1}{n+1+2k}} \cdot \frac{1}{2}^{m-\frac{n-1}{n+1+2k}} \cdot$$

$$\cdot \left[n+1+2k\right]^{\frac{n-1}{2}} \; n^{\frac{1}{4}} \; 2^{\frac{n}{2}} \; \left(k+1+\tfrac{n}{2}\right)^{-\frac{1}{2}\frac{n-1}{n+1+2k}}.$$

Now we proceed to the random variable $\sum_{BS \in F_1} s(h_{BS})$.

The philosophy here is as follows. We want to know how many points have to be checked in the total process. For each facet, which is to be determined, the set of checked points may differ.
So for each facet we count the number of checks and summarize. Formally this can be seen in the following way.
One enumerates all $\binom{m}{n}$ simplices, determines their heights and counts all points not belonging to Δ, which have a Euclidean length greater than h. But then, we ignore this result if the simplex does not have the facet-property. So our candidate enumeration works as follows:
We have $\binom{m}{n}$ simplices of the type $CH(a_{\Delta^1},\ldots,a_{\Delta^n})$ and we have $m-n$ additional points a_j ($j \notin \Delta$), which could be checked. So there are $\binom{m}{n}(m-n)$ combinations. The combination must be counted if the simplex is a boundary simplex and if $\|a_j\| \geq h$. For formalizing that count-decision, we rely on indicator functions.

$$(6.17)\quad E_{m,n}^{k}\left(\sum_{BS \in F_1} s(h_{BS})\right) = \binom{m}{n}(m-n) \cdot$$

$\cdot\ P\Big(CH(a_{\Delta^1},...,a_{\Delta^n})$ is facet of first kind and $\|a_j\| \wedge h(a_1,...,a_n)\Big) =$

$$= \binom{m}{n}(m-n) \int_{\mathbb{R}^n}\cdots\int_{\mathbb{R}^n} G\big(h(a_1,...,a_n)\big)^{m-n-1} \cdot$$

$$\cdot\ I_{a_j \text{ below } H(a_1,...,a_n) \text{ and } \|a_j\| \wedge h(a_1,...,a_n)}\Big(a_j, h(a_1,...,a_n), H(a_1,...,a_n)\Big) \cdot$$

$$\cdot\ d\hat{F}(a_j)\ d\hat{F}(a_1)\ldots d\hat{F}(a_n).$$

The integral becomes much simpler, if we increase the right side (by a factor at most 2) through dropping the condition that "a_j below $H(a_1,...,a_n)$".

$$(6.18)\ E^k_{m,n}\left(\sum_{BS\in F_1} s(h_{BS})\right) \le \binom{m}{n}(m-n) \int_{\mathbb{R}^n}\cdots\int_{\mathbb{R}^n} G\big(h(a_1,...,a_n)\big)^{m-n-1} \cdot$$

$$\cdot\ I_{\|a_j\| \wedge h}(a_j, h) \cdot d\hat{F}(a_j)\ d\hat{F}(a_1)\ldots d\hat{F}(a_n) =$$

with the methods of [Borgwardt (1987)] one gets to

$$= \binom{m}{n}(m-n)\,\lambda_{n-1}(\omega_n) \int_0^1 G(h)^{m-n}\ \Lambda_1(h)\,(1-F(h))\,dh.$$

New is only the factor $1-F(h)$, which specializes to

$$(6.19)\ 1-F_k(h) = \frac{\int_h^1 (1-\tau^2)^k \tau^{n-1} d\tau}{\int_0^1 (1-\tau^2)^k \tau^{n-1} d\tau} = \frac{\int_h^1 (1-\tau^2)^k \tau^{n-1} d\tau}{\dfrac{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}{2\Gamma\left(k+1+\frac{n}{2}\right)}} \le$$

$$\le \frac{\int_h^1 (1-\tau^2)^k \tau\, d\tau}{\dfrac{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}{2\Gamma\left(k+1+\frac{n}{2}\right)}} = \frac{\frac{1}{2}\frac{1}{k+1}\left(1-h^2\right)^{k+1}}{\dfrac{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}{2\Gamma\left(k+1+\frac{n}{2}\right)}} = \frac{\frac{1}{k+1}\left(1-h^2\right)^{k+1}}{\dfrac{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}}.$$

Insertion into (6.18) delivers

$$(6.20)\quad E^k_{m,n}\left(\sum_{BS\in F_1} s(h_{BS})\right) \le \binom{m}{n}(m-n)\,\lambda_{n-1}(\omega_n)\int_0^1 G_k(h)^{m-n-1}\; g_{0,k}(h)\;\cdot$$

$$\cdot\;(1-h^2)^{\left[\frac{n}{2}+k\right](n-1)+k+1}\; dh\;\cdot\;[n!]^{\frac{1}{2}}\left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}}\;\cdot$$

$$\cdot\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}}\;\cdot\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{(k+1)\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}\;=$$

$$=\binom{m}{n}(m-n)\,\lambda_{n-1}(\omega_n)\int_0^1 G_k(h)^{m-n-1}\; g_{0,k}(h)\left[1-G_k(h)\right]^{\frac{\left[\frac{n}{2}+k\right](n-1)+k+1}{\frac{n+1}{2}+k}}\; dh\;\cdot$$

$$\cdot\;[n!]^{\frac{1}{2}}\left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}}\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}}\;\cdot$$

$$\cdot\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{(k+1)\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}\;\left[\frac{2\sqrt{\pi}\;\;\Gamma\left(k+1+\frac{n-1}{2}\right)\left(\frac{n+1}{2}+k\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{\frac{\left[\frac{n}{2}+k\right](n-1)+k+1}{\frac{n+1}{2}+k}}\;=$$

$$=\binom{m}{n}(m-n)\,\lambda_{n-1}(\omega_n)\int_0^1 G_k(h)^{m-n-1}\;\; g_{0,k}(h)\left[1-G_k(h)\right]^{n-1+\frac{2k+3-n}{n+1+2k}}\; dh\;\cdot$$

$$\cdot\;[n!]^{\frac{1}{2}}\left(\frac{1}{2\pi}\right)^{\frac{n-1}{2}}\;\cdot\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)^{n-1}}{\Gamma\left(k+1+\frac{n-1}{2}\right)^{n-1}\left(k+1+\frac{n-1}{2}\right)^{\frac{n-1}{2}}}\;\cdot$$

$$\cdot\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{(k+1)\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}\;\cdot\;\left[\frac{2\sqrt{\pi}\;\;\Gamma\left(k+1+\frac{n-1}{2}\right)\left(\frac{n+1}{2}+k\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{n-1+\frac{2k+3-n}{n+1+2k}}\;=$$

$$= \binom{m}{n} (m-n)\, \lambda_{n-1}(\omega_n) \int_{\frac{1}{2}}^{1} G^{m-n-1} \; [1-G]^{n-1+\frac{2k+3-n}{n+1+2k}} \; dG \;\cdot$$

$$\cdot\; [n!]^{\frac{1}{2}} \;\; 2^{\frac{n-1}{2}} \;\; \left(\tfrac{n+1}{2}+k\right)^{\frac{n-1}{2}} \;\; \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{(k+1)\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \;\;\cdot$$

$$\cdot\; \left[\frac{2\sqrt{\pi}\;\; \Gamma\left(k+1+\frac{n-1}{2}\right)\left(\frac{n+1}{2}+k\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{\frac{2k+3-n}{n+1+2k}} .$$

Again, we exploit an upper bound for the integral from [Borgwardt (1987)]. Here we distinguish the cases where $\delta = \dfrac{n-3-2k}{n+1+2k}$ is positive or negative.

1) $$\binom{m}{n} (m-n) \int_0^1 G^{m-n-1} \; [1-G]^{n-1-\delta} \, dG \;\leq \qquad (\delta \;\; \text{positive})$$

$$\leq\; (m-n) \cdot \frac{1}{n} \left(\frac{m+1-\delta}{n-1}+k\right)^{\delta} \;\leq\; \left(\frac{1}{n-1}\right)^{1+\delta} (m+1)^{1+\delta}$$

2) $$\binom{m}{n} (m-n) \int_0^1 G^{m-n-1} \; [1-G]^{n-1-\delta} \, dG \;=\qquad (\delta \;\; \text{negative})$$

$$=\; (n+1) \binom{m}{n+1} \int_0^1 G^{m-(n+1)} \; [1-G]^{n-(1+\delta)} \, dG \;\leq$$

$$\leq\; \left(\frac{m+1-(1+\delta)}{n}\right)^{(1+\delta)} \;=\; \left(\frac{m-\delta}{n}\right)^{1+\delta} \;\leq$$

$$\leq\; \left(\frac{m+1}{n}\right)^{1-|\delta|}$$

Hence we have a common upper bound of

$$\left(\frac{1}{n-1}\right)^{1+\delta}(m+1)^{1+\delta}.$$

Insertion yields

$$(6.21)\quad E_{m,n}^{k}\left(\sum_{BS\in F_1} s(h_{BS})\right) \;\leq\; (m+1)^{1+\frac{n-3-2k}{n+1+2k}}\left(\frac{1}{n-1}\right)^{1+\frac{n-3-2k}{n+1+2k}} \cdot$$

$$\cdot\;\frac{2\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}\right)}\;[n!]^{\frac{1}{2}}\;2^{\frac{n-1}{2}}\;\left(\frac{n+1}{2}+k\right)^{\frac{n-1}{2}}\;\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{(k+1)\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)}\;\cdot$$

$$\cdot\;\left[\frac{2\sqrt{\pi}\;\;\Gamma\left(k+1+\frac{n-1}{2}\right)\left(\frac{n+1}{2}+k\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{\frac{2k+3-n}{n+1+2k}}.$$

Now we arrive at the last expectation value, namely the number of points which have to be sorted. Remember that in the case of nonexistence of facets of second kind a point needs to be in the subset of sorted points only if there is at least one facet of first kind whose height is smaller than the Euclidean length of the point. Since it requires (ln *m)*-time to determine the next greatest element of a set, when a heap is installed, the total effort for sorting the critical subset is (ln m) $s(Min^{\geq 0}\ h_{\mathsf{F}})$, where $Min^{\geq 0}\ h_{\mathsf{F}}$ describes the minimal height of a facet of Y and facets of second kind are given "height 0".

In order to obtain a practicable estimation, we must use a seemingly crude method.
For an appropriate $\bar{h}(m,n,k)$ we suggest that all points with $\|a_i\| > \bar{h}$ are sorted anyway.
Since $Min^{\geq 0}\ h_{\mathsf{F}}$ is a very complicated and from its distribution hardly computable random variable, we are going to estimate in the remaining area as follows:

$$(6.22)\ E(\#\{a_i \mid Min^{\geq 0}\ h_{\mathsf{F}} < \|a_i\| < \bar{h}\}) \;\leq\; E\left(\#\left\{BS \,\middle|\, h_{BS} < \bar{h}\right\}\right)\cdot(m-n).$$

Here we count a point as often as there is a facet of smaller height than $\bar{h}$.
However, this delivers an upper bound.

(6.23) $E^k_{m,n}\ \left(s\left(Min^{\wedge 0}\ h_{\mathrm{F}}\right)\right) \ \le\ m\left[1-F(\overline{h})\right]\ +$

$$+\ \binom{m}{n}\,(m-n)\,\lambda_{n-1}(\omega_n)\int_0^{\overline{h}} G(h)^{m-n}\ \ \Lambda_1(h)\,dh\ .$$

Since we know that

(6.24) $\lambda_{n-1}(\omega_n)\ \int_0^1 \Lambda_1(h)\,dh\ =\ 1$ (#F_1 when $m=n$)

we can conclude that

(6.25) $\binom{m}{n}\,(m-n)\,\lambda_{n-1}(\omega_n)\int_0^{\overline{h}} G(h)^{m-n}\ \ \Lambda_1(h)\,dh\ \le$

$$\le\ \binom{m}{n}\,(m-n)\ G(\overline{h})^{m-n}\ \ \lambda_{n-1}(\omega_n)\ \int_0^1 \Lambda_1(h)\,dh\ \le$$

$$\le\ m^{n+1}\ G(\overline{h})^{m-n}\,.$$

Now we try to find a $\overline{h}$, which simultaneously keeps (6.25) and $m(1-F_k(\overline{h}))$ relatively small.
Let us consider the case where $m >> n$. Then choose $\overline{h}$ such that

(6.26) $(1-\overline{h}^2)\ =\ \left[\dfrac{\ln m}{m-n}\sqrt{\pi}\,(n+1)\,(n+1+2k)\right]^{\frac{1}{\frac{n+1}{2}+k}}$

(This becomes < 1 for sufficiently great m).

Then (compare (6.19))

(6.27) $m[1-F_k(\overline{h})]\ \le\ \dfrac{\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)(k+1)}\ \ m\ \cdot$

$$\cdot \left[\frac{\ln m}{(m-n)} \sqrt{\pi}\,(n+1)(n+1+2k)\right]^{\frac{k+1}{\frac{n+1}{2}+k}} .$$

And the result of (6.25) can be bounded by (6.10)

$$(6.28)\ G_k(\bar{h}) \leq 1 - \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}\ (1-\bar{h}^2)^{\frac{n+1}{2}+k} .$$

Then

$$(6.29)\ m^{n+1}\ G(\bar{h})^{m-n} \leq e^{(\ln m)(n+1)}\ \cdot$$

$$\cdot \left[1 - \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)} \cdot \left(\frac{\ln m}{(m-n)} \sqrt{\pi}\,(n+1)\,(n+1+2k)\right)^{\frac{\frac{n+1}{2}+k}{\frac{n+1}{2}+k}}\right]^{m-n} =$$

$$= e^{(\ln m)(n+1)} \left[1 - \frac{\ln m}{m-n}\ \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}\ \sqrt{\pi}\ (n+1)\,(n+1+2k)\right]^{m-n} \leq$$

$$\leq e^{(\ln m)(n+1)}\ \exp\left\{-\ln m(n+1)\cdot \frac{\Gamma\left(k+1+\frac{n+2}{2}\right)}{\Gamma\left(k+1+\frac{n+1}{2}\right)}\right\} =$$

$$= \exp\left\{\ln m(n+1)\cdot\left[1 - \frac{\Gamma\left(k+1+\frac{n+2}{2}\right)}{\Gamma\left(k+1+\frac{n+1}{2}\right)}\right]\right\} \leq 1 .$$

Now (6.27) delivers an upper bound for the order of the effort for sorting

$$(\ln m)\, m\ \cdot \left(\frac{1}{m-n}\right)^{\frac{k+1}{\frac{n+1}{2}+k}} \left[\ln m\ \sqrt{\pi}\,(n+1)\,(n+1+2k)\right]^{\frac{k+1}{\frac{n+1}{2}+k}} \cdot$$

$$\cdot \ \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\,\Gamma\left(\frac{n}{2}\right)(k+1)} \qquad \text{in the case } m >> n.$$

This shows that the effort for sorting remains linear in m in the asymptotical case.

7. The main results

We summarize all our expectation values, insert them in theorem 5 and describe the total effort.

Theorem 6

Let $a_1,\ldots,a_m$ be distributed independently, identically and symmetrically under rotations with the radial density specified by the parameter k [$(-1, \infty)$

$$f_k(r) = \frac{(1-r^2)^k \, r^{n-1}}{\int_0^1 (1-\tau^2)^k \, \tau^{n-1} d\tau} \qquad \text{for } 0 \le r \le 1 \qquad \text{and } 0 \text{ for } r > 1.$$

Then the average number of arithmetic operations for determining the convex hull can be estimated from above by

$$C \Big\{ n^3 + n^2 m + n^3 \ln m +$$

$$+ \; n^2 C_1(n,k) \left(m^{\frac{n-1}{n+1+2k}} + \ln m \right) +$$

$$+ \; n \, C_2(n,k) \; (m+1)^{1+\frac{n-3-2k}{n+1+2k}} +$$

$$+ \; n \, C_3(n,k) \binom{m}{n} \frac{n+m+n\ln m}{m - \frac{n-1}{n+1+2k}} \; \frac{1}{2}^{\,m-\frac{n-1}{n+1+2k}} +$$

$$+ \; m \; + \; (\ln m)\, m.^{*)} \Big\}$$

*) For $m \gg n$ we can replace $(\ln m)\, m$ by

$$(\ln m)^{\frac{n+1+4k+2}{n+1+2k}} \; m^{\frac{n+1-2}{n+1+2k}} \cdot C_4(n,k).$$

All these figures $C_i(n,k)$ can be given explicitly

$$C_1(n,k) = \frac{1}{n}\, 2\pi^{\frac{n}{2}}\, \frac{[n!]^{\frac{1}{2}}}{\Gamma\left(\frac{n}{2}\right)}\, (n+1+2k)^{\frac{n-1}{2}} \cdot \left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}\right]^{\frac{n-1}{n+1+2k}} .$$

$$C_2(n,k) = \left(\frac{1}{n-1}\right)^{1+\frac{n-3-2k}{n+1+2k}} \cdot 2^{\frac{n+1}{2}}\, \pi^{\frac{n}{2}}\, \frac{[n!]^{\frac{1}{2}}}{\Gamma\left(\frac{n}{2}\right)} \cdot \left(\frac{n+1}{2}+k\right)^{\frac{n-1}{2}} \cdot$$

$$\cdot \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{(k+1)\,\Gamma(k+1)\Gamma\left(\frac{n}{2}\right)} \cdot \left[\frac{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}{\Gamma\left(k+1+\frac{n}{2}\right)}\right]^{\frac{2k+3-n}{n+1+2k}} .$$

$$C_3(n,k) = \frac{2\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}\right)}\, [n!]^{\frac{1}{2}}\, [n+1+2k]^{\frac{n-1}{2}} \cdot \left[\frac{\Gamma\left(k+1+\frac{n}{2}\right)}{2\sqrt{\pi}\ \Gamma\left(k+1+\frac{n+1}{2}\right)}\right]^{\frac{n-1}{n+1+2k}} .$$

$$C_4(n,k) = \left[\sqrt{\pi}\,(n+1)\,(n+1+2k)\right]^{\frac{k+1}{\frac{n+1}{2}+k}} \cdot \frac{\Gamma\left(k+1+\frac{n}{2}\right)}{\Gamma(k+1)\,\Gamma\left(\frac{n}{2}\right)(k+1)} .$$

Apparently the m-dependency is dominated by the result from $\sum_{BS\in F_1} s(h_{BS})$ of

$$(m+1)^{1+\frac{n-3-2k}{n+1+2k}} = (m+1)^{2\frac{(n-1)}{n+1+2k}} = (m+1)^{2-\frac{4+4k}{n+1+2k}} .$$

This holds as long as $k < \dfrac{n-3}{2}$. After that the sorting term and heap effort become the strongest.

Theorem 7

For our special cases we obtain the following dependency on m ($m \to \infty$, n fixed)

1) $k \to -1$ (uniform distribution on ω_n) $\quad m^{2\frac{n-1}{n+1+2k}} \underset{k\to -1}{\longrightarrow} m^2$

2) $k = 0$ (uniform distribution on Ω_n) $m^{2\frac{n-1}{n+1}} = m^{2-\frac{4}{n+1}}$

3) $k > \frac{n-3}{2}$ (distribution centrally directed) m

$k = \frac{n-1}{2}$ (symmetric radial density) m

4) $k \to \infty$ (totally centralized) m.

The question remains, whether the Throw-Away-Principle could also be combined with Gift-Wrapping without storage. But this seeems to be inefficient in most cases. Remember that Avis & Fukuda work with a worst-case-bound of $O(m \cdot n \cdot \#\mathrm{F})$.
If we want to implement the Throw-Away-idea, we must calculate for each point-check the potential height of the preliminary facet. But Avis & Fukuda make the evaluation of all rows (instead of one row) necessary, because every row is a candidate for a reverse pivot step and no other information (storage) is available. So we would have to accept another factor n in the main term of our result, which means

$$E(n^2 \sum_{BS \in \mathrm{F}} s(h_{BS})) \quad \text{for the combination instead of}$$

$$E(n \sum_{BS \in \mathrm{F}} s(h_{BS})) \quad \text{for pure Throw-Away-Principle.}$$

This drawback has to be weighted against the advantage of saving of the storage effort of $(\#\mathrm{F}) \cdot n^2 \ln m$.
So, in almost all cases the combination is inferior to the pure Throw-Away-Principle.

References

D. Avis and K. Fukuda (1992): A Pivoting Algorithm for Convex Hulls and Vertex Enumeration of Arrangements and Polyhedra. *Discrete Computational Geometry* **8**, 295-313.

J. Bentley and M. Shamos (1978): Divide and Conquer for Linear Expected Time. *Information Processing Letters* **7, Nr.2**, 87-91.

K. H. Borgwardt (1987): The Simplex Method. In: *Algorithms and combinatorics* (Eds. Graham, Korte and Lovász). Berlin, Springer.

K. H. Borgwardt, N. Gaffke, M. Jünger and G. Reinelt, Eds. (1991): *Computing the Convex Hull in the Euclidean Plane in Linear Expected Time*. DIMACS
Applied Geometry and Discrete Mathematics
The Victor Klee Festschrift.

C. Buchta, Eds. (1985): *Zufällige Polyeder - eine Übersicht*. Lecture Notes in Mathematics. Berlin, Springer.

P. L. Chand and S. S. Kapur (1970): An Algorithm for Convex Polytopes. *Journal of the Association for Computing Machinery* **17**(1), 78-86.

K. Clarkson (1986): Linear programming in O(n 3^{dxd})time. *Info. Proc. Letters* **22(1)**, 21-24.

L. Devroye (1983): Moment Inequalities for Random Variables in Computational Geometry. *Computing* **30**, 111-119.

R. A. Dwyer (1989). Random Convex Hulls in a Product of Balls. North Carolina State University.

R. A. Dwyer (1988a). Average-Case Analysis of Algorithms for Convex Hulls and Voronoi Diagrams. Carnegie Mellon-University, Computer Science Dept.

R. A. Dwyer (1988b): On the Convex Hull of Random Points in a Polytope. *Journal of Applied Probability* **25**, 688-699.

M. Dyer and A. Frieze (1989): A Randomized Algorithm for Fixed-Dimensional Linear Programming. *Math. Programming* **44**, 203-212.

M. E. Dyer (1984): Linear Time Algorithms for Two-And Three-Variable Linear Programs. *SIAM Journal of Computing* **13**(1), 31-45.

R. L. Graham (1972): An Efficient Algorithm for Determining the Convex Hull of a Finite Planar Set. *Information Processing Letters* **1**, 132-133.

M. Jünger and G. Reinelt (1989). Fast Computation of Convex Hulls. Universität Augsburg Schwerpunktprogramm der DFG:
Anwendungsbezogene Optimierung und Steuerung.

K. Küfer (1988): *Zur Asymptotik in der stochastischen Polyedertheorie*. Diplomarbeit, Universität Kaiserslautern.

K. Küfer (1992): *Asymptotische Varianzanalysen in der stochastischen Polyedertheorie*. Doktorarbeit, Universität Kaiserslautern.

N. Megiddo (1984): Linear Programming in Linear Time when the Dimension is Fixed. *J. Assoc. Comput. Mach.* **31**, 114-127.

R. Seidel (1991): Small-Dimensional Linear Programming and Convex Hulls Made Easy. *Discrete Computational Geometry* **6**, 423-434.

G. Swart (1985): Finding the Convex Hull Facet by Facet. *Journal of Algorithms* **6**, 17-48.

B. Van Wel (1989): The Convex Hull of a Uniform Sample from the Interior of a Simple d-Polytope. *J. Appl. Prob.* **26**, 259-273.